
Rule WLM132: Significant transaction time was in Waiting (Miscellaneous) state

Finding: A significant amount of the transaction response time for the service class missing its performance goal was spent in the Waiting (Miscellaneous) state. This finding applies to service classes which are part of a subsystem (e.g., CICS transactions).

Impact: This finding has MEDIUM IMPACT or HIGH IMPACT on performance of the service class. The level of impact depends on the percent of transaction response time spent in the Waiting (Miscellaneous) state.

Logic flow: The following rules cause this rule to be invoked:
 Rule WLM104: Subsystem Service Class did not achieve average response goal
 Rule WLM105: Subsystem Service Class did not achieve percentile response goal

Discussion: When CPExpert produces Rule WLM104 or Rule WLM105 to indicate that a subsystem service class did not achieve its performance goal, the logic of these rules tries to identify the cause of the delay. The cause of the delay initially is analyzed from the "served" service class view. The delays from the served service class are reported by CICS/ESA Version 4.1 or IMS Version 5 interaction with the Workload Manager, using the Workload Management Services macros¹.

CICS/ESA Version 4.1 reports two separate views of the transactions: the *begin_to_end phase* and the *execution phase*².

- **Begin_to_end phase.** The begin_to_end phase starts when CICS/ESA Version 4.1 has classified the transaction³. This action normally is done in a CICS Terminal Owning Region (TOR).
- **Execution phase.** The execution phase starts when either CICS/ESA Version 4.1 or IMS Version 5 has started an application task to process

¹Please refer to Section 4 of this document for more detail about the Workload Management Services macros and how the subsystems use these macros to exchange information with the Workload Manager.

²IMS Version 5 reports only *execution phase* samples.

³Classifying the transaction into a service class is actually done by the Workload Manager when CICS issues the IWMCLSFY macro. Please refer to Section 4 for a more complete discussion of the subsystem work manager (e.g., CICS) interaction with the Workload Manager.

the transaction. For CICS, this normally is done in a CICS Application Owning Region (AOR).

Within each phase, CICS or IMS reports the "state" of the transaction, from the view of CICS or IMS. The state of the transaction is reported in the following categories⁴:

- **Idle state.**
- **Active state.**
- **Ready state.**
- **Wait state.**
- **Switched state.**

If the subsystem supports work manager delay reporting, the delay information is available in the "Work Manager/Resource Manger State Section" of SMF Type 72 (Subtype 3) records. When a transaction service class fails to achieve its performance goal, CPExpert analyzes the information to identify the primary and secondary causes of delay.

The Wait state indicates that a task in support of the transaction was waiting on some activity. The Wait state is broken into several categories: waiting for lock, waiting for I/O, waiting for conversation, waiting for distributed request, waiting for a session to be established (locally, somewhere in the network, or somewhere in the sysplex), waiting for a timer, waiting for another product, or waiting for an unidentified resource.

CICS reports the time when a work unit (that is, a task in support of a transaction) was waiting, broken into ten separate categories. Nine of the waiting categories are specific (e.g., Waiting for I/O). The tenth category is the "Miscellaneous Wait" category, used when CICS does not identify the specific reason for the wait delay.

The initial versions of CICS documentation simply described the "Miscellaneous Wait" category as being wait for unidentified reasons. In revisions to the documents, IBM has provided detailed information about the Workload Manager Miscellaneous Wait types used by CICS.

⁴Please refer to Section 4 of this document for a more comprehensive discussion of the transaction states and the interaction between the subsystem (CICS or IMS) and the Workload Manager.

CPEXpert produces Rule WLM132 when the primary or secondary cause of delay was that the transaction service class was in the Waiting (Miscellaneous) state for a significant percent of its response time.

The following example illustrates the output from Rule WLM132:

```
RULE WLM132:  SIGNIFICANT TRANSACTION TIME WAS WAITING, MISCELLANEOUS
```

```
A significant amount of the transaction response time for the CICSPROD  
Service Class was spent waiting for reasons which were not identified  
by CICS. Please refer to the description of Rule WLM132 for a discussion  
of the CICS Miscellaneous Wait categories, how to determine which CICS  
Miscellaneous Waits occur on your system, and how to reduce these waits.
```

Suggestion: IBM has provided detailed information about the Workload Manager Miscellaneous Wait types used by CICS. Exhibit WLM132-1 shows the resources that a suspended task might be waiting on for the Workload Manager Miscellaneous Wait type.

As shown in Exhibit WLM132-1, there are twelve reasons that CICS provides the Workload Manager with a Miscellaneous Wait.

- **CICS system task waits.** CICS system task waits occur (1) as a natural result of the CICS system tasks or (2) because of a system error preventing the system task from resuming.
- Many system tasks enter a wait state as a natural result of their operation.
- For example, the DFHSMYSY module of the storage manager domain might stay suspended for a prolonged time (i.e., minutes, or even hours). The purpose of the DFHSMYSY module is to clean up storage when significant changes occur in the amount being used. This situation would happen infrequently in a production system running well within its planned capacity, but the situation can occur.
- Some system tasks perform many I/O operations. These I/O operations are subject to I/O constraints such as string availability, and volume and data set locking. In the case of tape volumes, the tasks can also be dependent on operator action while new volumes are mounted.

TYPE OF WAIT	TYPE OF TASK	RESOURCE TYPE	RESOURCE NAME	SUSPENDING MODULE
CICS system task waits	System task	(none)	DMWTQUEU	DFHDMWQ
CICS system task waits	System task	AP_INIT	CSADLECB	DFHSII1
CICS system task waits	System task	AP_INIT	ECBTC	DFHAPSIP
CICS system task waits	System task	AP_INIT	SIPDMTEC	DFHAPSIP
CICS system task waits	System task	AP_INIT	TCTVCECB	DFHSII1
CICS system task waits	System task	AP_QUIES	CSASSI2	DFHSTP
CICS system task waits	System task	AP_QUIES	SHUTECB	DFHSTP
CICS system task waits	System task	DBDXEOT	(none)	DFHDXSTM
CICS system task waits	System task	DBDXINT	(none)	DFHXSTM
CICS system task waits	System task	DFHAIIN	AITM	DFHAIIN1
CICS system task waits	System task	DFHCPIN	CPI	DFHCPIN1
CICS system task waits	System task	DFHPRIN	PRM	DFHPRIN1
CICS system task waits	System task	DFHSIPLT	EARLYPLT	DFHSII1
CICS system task waits	System task	FCINWAIT	STATIC	DFHFCIN1
CICS system task waits	System task	JCINITN	JOURNALS	DFHJCP
CICS system task waits	System task	STARTUP	TSMCECB	DFHRCRP
CICS system task waits	System task	SUBTASK	SISUBECB	DFHRCRP
CICS system task waits	System task	SUCNSOLE	WTO	DFHSUWT
CICS system task waits	System task	TCP_SHUT	DFHZDSP	DFHZDSP
EDF waits	User task	EDF	DEBUGUSER	DFHEDFX
Front End Programming waits	User task	ADAPTER	FEPI_RQE	DFHSZATR
Front End Programming waits	CSZI	FEPRM	SZRDP	DFHSZRDP
Interval control waits	User task	ICGTWAIT	terminal_ID	DFHICP
Interval control waits	User task	ICWAIT	terminal_ID	DFHICP
Journal control waits	User task	JASUBTAS	JASTMECB	DFHJCSDJ
Journal control waits	User task	JCBUFFER	JCTBAECB	DFHJCSDJ
Journal control waits	User task	JCDETACH	SUBTASK	DFHJCSDJ
Journal control waits	User task	JCREADY	JCTXAECB	DFHJCO
Journal control waits	User task	JCREADY	JCTXBECB	DFHJCO
Journal control waits	User task	JCREADY	JCTXXECB	DFHJCO
Storage waits	User task	CDSA	(none)	DFHSMSQ
Storage waits	User task	ECDSA	(none)	DFHSMSQ
Storage waits	User task	ERDSA	(none)	DFHSMSQ
Storage waits	User task	ESDSA	(none)	DFHSMSQ
Storage waits	User task	ESDSA	(none)	DFHSMSQ
Storage waits	User task	EUDSA	(none)	DFHSMSQ
Storage waits	User task	RDSA	(none)	DFHSMSQ
Storage waits	User task	SDSA	(none)	DFHSMSQ
Storage waits	User task	UDSA	(none)	DFHSMSQ
Task control waits	User task	EKCWAIT	SINGLE	DFHEKC
Task control waits	User task	KCCOMPAT	LIST	DFHXCFA
Task control waits	User task	KCCOMPAT	SINGLE	DFHXCFA
Task control waits	User task	KCCOMPAT	SUSPEND	DFHXCFA
Task control waits	User task	KCCOMPAT	TERMINAL	DFHXCFA
Temporary storage wait	User task	TSAX	(none)	DFHTSP
Transient data waits	User task	TD_INIT	DCT	DFHTDA
User waits	User task	FOREVER	DFHXMTA	DFHXMTA
User waits	User task	USERWAIT	ECB	list
VTAM waits	User task	ZCIOWAIT	DFHZARER	DFHZARER
VTAM waits	User task	ZCZGET	DFHZARL2	DFHZARL
VTAM waits	User task	ZCZNAC	DFHZARL3	DFHZARL
XRF waits	User task	XRPUTMSG	message_Q	DFHWMQP

CICS MISCELLANEOUS WAITS

Exhibit WLM132-1

You should consider placing CICS system tasks into a single service class. IBM suggests that you not mix CICS-supplied transactions in a service class with user transactions.

- You should contact your IBM support center if a system task is in a wait state, and there is a system error preventing it from resuming.
- **Execution Diagnostic Facility (EDF) waits.** The EDF waits are a natural result of using the Execution Diagnostic Facility.

The EDF waits should not occur in a CICS production region. EDF waits would not be a cause for concern in a CICS test region, as they are programmer-generated.

- **Front End Programming waits.** There are two types of Front End Programming waits from the view of CICS: (1) a wait for the FEPI_RQE resource and (2) a wait for the SCRDP resource.

- The wait for the FEPI_RQE resource is issued in the FEPI adapter when a FEPI command is passed to the Resource Manager for processing. The Wait ends when the Resource Manager has processed the request. It is possible for a FEPI_RQE wait to be outstanding for a long time (for example, when awaiting a flow from the back-end system that is delayed due to network traffic). IBM recommends that you not cancel tasks that are waiting at this point; to do so could lead to severe application problems.
- The wait for the SCRDP resource is issued by the CSZI task in the FEPI Resource Manager when it has no work to do. The wait ends when work arrives (from either the FEPI adapter or a VTAM exit).

An SZRDP wait is generated when the FEPI Resource Manager is idle. Consequently, the SZ TCB is also inactive. On lightly loaded systems, this occurs frequently.

The Dispatcher Domain Statistics part of the CICS interval statistics contain information which can be analyzed to determine whether the WLM Miscellaneous Wait was likely caused by a Front End Programming wait. There are Dispatcher Domain Statistics for each TCB; TCB 4 is the secondary LU TCB and is present if FEPI=YES was specified in the System Initialization Table. Within TCB 4 statistics, the DSGTWT field holds the accumulated real time that the CICS region was in a MVS wait for the Front End Programming TCB.

If the DSGTWT value is small, you can be reasonably sure that the WLM Miscellaneous waits were **not** caused by Front End Programming waits.

If the DSGTWT value is relatively large, it is possible that the WLM Miscellaneous waits **were** caused by Front End Programming waits. Unfortunately, there is no way to determine whether a task suspended for a Front End Programming Wait actually was in the service class missing its performance goal. However, **some** tasks in the CICS region are encountering Front End Programming Waits if the DSGTWT value is relatively large and you may wish to take action.

The CICS/ESA Front End Programming Interface User Guide (see References) should be consulted regarding improving the performance of the Front End Programming interface.

Additionally, you should consider placing CICS system tasks into a single service class. IBM suggests that you not mix CICS-supplied transactions in a service class with user transactions.

- **Interval Control waits.** Interval Control waits are caused by user tasks.

You should review the "Interval Control Waits" part of Section 2.3: Dealing with waits (Bookmanager document) of the CICS/ESA Version 4.1 Problem Determination Guide.

- **Journal Control waits.** CICS Journal Control provides the Workload Manager with a Miscellaneous Wait for four resource types: JASUBTAS, JCBUFFER, JCDETACH, and JCREADY.
 - **JASUBTAS.** The purpose of the wait for the JASUBTAS resource is to delay shutdown until the JASP subtask has completely submitted all the archiving jobs of those journals needing to be archived.
 - **JCBUFFER.** If the resource type is JCBUFFER, with resource name JCTBAECB, the task that has requested shutdown is waiting for the journaling task to flush the buffer, close the journal, and terminate itself.
 - **JCDETACH:** A task that has requested shutdown can be made to wait on the detaching of the journal subtask from the operating system.
 - **JCREADY.** Workload Manager Miscellaneous Waits for the JCREADY resource type occur during archiving. CICS writes to a second data set while archiving the first data set either tape or disk. The first data set is not reused until archiving is complete and the operator has responded to message DFHJC4583. If the operator has not responded before the second journal data set is full, the JCT PAUSE option causes logging to cease until the operator has

responded. User tasks are made to wait on resource type JCREADY when no operator reply has been received to message DFHJC4583, and message DFHJC4584 has subsequently been issued.

Workload Manager Miscellaneous Waits for the first three Journal Control resource types occur only during shutdown, and should not cause a service class to miss its performance goal.

Workload Manager Miscellaneous Waits for the JCREADY resource type could cause serious performance problems if the operator does not respond to message DFHJC4583 in a timely manner.

The Journal Control Statistics part of the CICS interval statistics contain information which can be analyzed to determine whether the WLM Miscellaneous Wait was likely caused by CICS having to wait for the archive job. The field A13WAC is a count of the number of times CICS had to wait for a particular journal because the archive job had not completed at the time it was needed.

- If the A13WAC field is zero, you can be sure that the WLM Miscellaneous waits were **not** caused by Journal Control archiving.
- If the A13WAC value is non-zero, CPExpert suggests that you determine why message DFHJC4583 was not responded to in a timely manner. While it is uncertain that the operator response caused problems with the service class missing its performance goal, tasks are suspended because of archiving problems. You should take action to correct the problem.

Alternatively, you can execute the CICS Component of CPExpert against the CICS region(s) serving the service class missing its performance goal. The CICS Component will analyze the CICS interval statistics to identify performance problems.

- **Storage waits.** Storage waits occur when a task is waiting for any of the resource types CDSA, UDSA, ECDSA, EUDSA, ERDSA, SDSA, ESDSA, or RDSA. Waits on these resources occur when tasks make unconditional storage requests (SUSPEND=YES) that cannot be satisfied⁵. Storage requests below the 16MB line wait for CDSA, UDSA, SDSA, or RDSA. Storage requests above the line 16MB line wait for ECDSA, EUDSA, ESDSA, or ERDSA.

⁵Note that, if conditional requests are made (SUSPEND=NO), tasks are not suspended on these resources, and a miscellaneous wait would not be provided to the Workload Manager.

CICS automatically takes steps to relieve storage when it is under stress. For example, CICS would release storage occupied by programs whose current use count is zero.

The most likely reasons for extended waits on storage requests are:

- The task has issued an unconditional GETMAIN request for an unreasonably large amount of storage.
- The task has issued an unconditional GETMAIN request for a reasonable amount of storage, but the CICS region is approaching a short-on-storage (SOS) condition.
- The task has issued an unconditional GETMAIN request for a reasonable amount of storage, but storage in the CICS region could have become too fragmented for the request to be satisfied.

The Storage Manager Statistics part of the CICS interval statistics contain information which can be analyzed to determine whether the WLM Miscellaneous Wait was likely caused by a storage wait. The field SMSUCSS is a count of the number of times a task was suspended because of insufficient storage to satisfy the request at the moment.

- If the SMSUCSS value is zero, you can be sure that the WLM Miscellaneous waits were **not** caused by storage waits.
- If the SMSUCSS value is non-zero, it is possible that the WLM Miscellaneous waits **were** caused by storage waits. Unfortunately, there is no way to determine whether a task suspended for storage constraint actually was in the service class missing its performance goal. However, tasks in the CICS region are encountering waits for storage if the SMSUCSS value is non-zero, and you should normally consider action. Further, the waiting task may be automatically purged⁶ if it has waited for storage longer than the deadlock time-out parameter specified in the installed transaction definition.

If the SMSUCSS value is non-zero, CPExpert suggests that you review the suggested actions beginning on page 171 of the IBM *CICS Version 4.1 Performance Guide*. These actions provide a checklist for reducing the virtual storage requirements above and below the 16MD line.

Alternatively, you can execute the CICS Component of CPExpert against the CICS region(s) serving the service class missing its performance

⁶Certain conditions prevent purging of a task (as examples, a deadlock time-out value of 0, or a specification of SPURGE(NO)).

goal. The CICS Component will analyze the CICS interval statistics to identify performance problems.

- **Task Control waits.** The CICS Transaction Manager provides the Workload Manager with a Miscellaneous Wait when a task is waiting on a resource type of KCCOMPAT, and the task has been suspended by the Transaction Manager. Additionally, CICS Task Control provides the Workload Manager with a Miscellaneous Wait when a task is waiting on a resource type of EKCWAIT and has been suspended by Task Control.
- The Miscellaneous Wait type is issued by the Transaction Manager when the task is suspended after issuing one of three macro calls:
 - A DFHKC TYPE=WAIT,DCI=LIST macro call was issued. The task is waiting for any ECB in a list of ECBs to be posted, after which the task may be resumed.
 - A DFHKC TYPE=WAIT,DCI=SINGLE macro call was issued. The task is waiting for a single ECB to be posted, after which the task may be resumed.
 - A DFHKC TYPE=WAIT,DCI=TERMINAL macro call was issued. CICS has suspended the task. The task is waiting for terminal I/O to complete, after which the task may be resumed.
- The Miscellaneous Wait type is issued by Task Control when the task is suspended on a resource type of EKCWAIT after issuing an EXEC CICS WAIT EVENT command. Task Control waits tend to be application-dependent. You should review the "Task Control Waits" part of Section 2.3: Dealing with waits (Bookmanager document) of the CICS/ESA Version 4.1 Problem Determination Guide.
- **Temporary Storage Waits.** Temporary storage is a scratchpad facility that is heavily used in many systems. Temporary storage exists in either main storage above the 16MB line (ECDSA), or auxiliary storage in a VSAM-managed data set. Temporary storage waits are related to temporary storage existing in auxiliary storage.

A task is forced to wait on temporary storage in auxiliary storage if the task has made an unconditional request for temporary storage, and the request cannot be met because insufficient auxiliary storage is available

There are two likely reasons why a task might be suspended waiting for temporary storage:

-
- The task has issued a request requiring too large a piece of temporary storage.
 - The task has issued a request requiring a reasonable amount of temporary storage, but there is too little available. This could indicate that the amount of auxiliary storage is becoming exhausted. Alternatively, there could be a relatively large amount of auxiliary storage available, but the storage is too fragmented for the request to be satisfied.

The Temporary Storage Statistics part of the CICS interval statistics contain information which can be analyzed to determine whether the WLM Miscellaneous Wait was likely caused by a Temporary Storage wait. The field A12STA8F field is a count of the number of times a task was suspended or had been abended because auxiliary storage had been exhausted.

- If the A12STA8F value is zero, you can be sure that the WLM Miscellaneous waits were **not** caused by Temporary Storage waits.
- If the A12STA8F value is non-zero, it is possible that the WLM Miscellaneous waits **were** caused by Temporary Storage waits. Unfortunately, there is no way to determine whether a task suspended for Temporary Storage constraint actually was in the service class missing its performance goal. However, tasks in the CICS region are encountering waits for Temporary Storage if the A12STA8F value is non-zero, and you should normally consider action. Further, the waiting task may be automatically purged⁷ if it has waited for temporary storage longer than the deadlock time-out parameter specified in the installed transaction definition. Otherwise, it is not purged, and is liable to be suspended indefinitely.

If the A12STA8F value is non-zero, CPExpert suggests that you review the suggested actions beginning on page 289 of the *IBM CICS Version 4.1 Performance Guide*. These actions provide a checklist for improving the performance of temporary storage residing on auxiliary storage.

Alternatively, you can execute the CICS Component of CPExpert against the CICS region(s) serving the service class missing its performance goal. The CICS Component will analyze the CICS interval statistics to identify performance problems.

⁷Certain conditions prevent purging of a task (as examples, a deadlock time-out value of 0, or a specification of SPURGE(NO)).

-
- **Transient Data waits.** Tasks issuing requests to read and write to transient data destinations can be suspended for several reasons. The reasons depend on the type of request being made, and whether the task is attempting to access an extrapartition or an intrapartition queue. One of the reasons a task is suspended is related to the TD_INIT resource type, and occurs during system initialization.

A second stage PLT program being executed during system initialization can issue a request for a resource that is not yet available, because the component that services the request has not yet been initialized. If the program issues a transient data request that cannot yet be serviced, it is suspended on a resource type of TD_INIT with a resource name of DCT. CICS provides the Workload Manager with a Miscellaneous Wait when a task is waiting on the TD_INIT resource type.

Workload Manager Miscellaneous Waits for Transient Data occur only during system initialization. These waits would not cause a service class to miss its performance goal because the region has not yet begun accepting transactions.

- **User waits.** CICS provides the Workload Manager with a Miscellaneous Wait when a task is waiting on an ECB list posted by the user. User waits are application dependent.
- **VTAM waits.** CICS provides the Workload Manager with a Miscellaneous Wait when a task is waiting on three resource types: ZCIOWAIT, ZCZGET, and ZCZNAC.
 - The ZCIOWAIT resource type wait is caused by a task waiting on terminal I/O.
 - The ZCZGET resource type wait is caused with application request logic for LU6.2 devices.
 - The ZCZNAC resource type wait is for DFHZNAC to issue an error message.
- **XRF alternate system waits.** CICS provides the Workload Manager with a Miscellaneous Wait when a task is waiting caused by XRF alternative system waits. The XRF takeover process is a major system event, and you would not expect individual tasks to perform well during the takeover.

To summarize the above discussion, the most likely causes of Workload Manager Miscellaneous Waits, during **normal** transaction processing, are:

(1) CICS system task waits, (2) storage waits, (3) temporary storage waits, and (4) application-dependent waits.

- You should consider placing CICS system tasks into a single service class. IBM suggests that you not mix CICS-supplied transactions in a service class with user transactions. Once this has been done, remaining waits are likely to be related to SUSPENDED user tasks.
- You can examine CICS interval statistics to determine whether the Miscellaneous Waits are related to storage waits or temporary storage waits. The preceding discussion describes the relevant fields in the interval statistics. Alternatively, you can execute the CICS Component of CPEXpert against the CICS region(s) serving the service class missing its performance goal. The CICS Component will analyze the CICS interval statistics to identify performance problems.
- If you have taken the above actions and Miscellaneous Waits remain a major cause of transaction delay during normal operations, the most likely cause is application-dependent waits. You may wish to examine applications to determine whether they cause the waits, or you may simply ignore the waits.

Reference: CICS/ESA Version 4.1 Performance Guide
Section 2.7.1.1: The response time breakdown in percentage section

CICS/ESA Version 4.1 Problem Determination Guide)
Section 2.3: Dealing with waits

CICS/ESA Front End Programming Interface User Guide)
Section 2.4.2 (Performance) - system-related performance
Section 3.4.5.2 (Performance) - application-related performance

CICS/TS Release 1.1 Performance Guide
Section 2.7.1.1: The response time breakdown in percentage section

CICS/TS Release 1.1 Problem Determination Guide)
Section 2.3: Dealing with waits

CICS/TS Release 1.2 Performance Guide
Section 2.7.1.1: The response time breakdown in percentage section

CICS/TS Release 1.2 Problem Determination Guide)
Section 2.3: Dealing with waits

CICS/TS Release 1.3 Performance Guide

Section 2.6.1.1: The response time breakdown in percentage section

CICS/TS Release 1.3 Problem Determination Guide)

Section 2.3: Dealing with waits

CICS/TS Front End Programming Interface User Guide

Section 2.4.2 (Performance) - system-related performance

Section 3.4.5.2 (Performance) - application-related performance

CICS/TS for z/OS Release 2.2 *Performance Guide*:

Chapter 8: Understanding RMF Workload Manager Data

CICS/TS for z/OS Release 2.2 Problem Determination Guide)

Chapter 6: Dealing with waits

CICS/TS for z/OS Release 2.2 Front End Programming Interface User Guide

Chapter 6: FEPI Performance

Chapter 14: Application Design (Performance)

Thanks:

Computer Management Sciences would like to recognize the efforts of the IBM CICS/ESA Development Team, IBM United Kingdom Laboratories (particularly Mr. Chris Baker) for providing detailed information about the resources that a CICS task might be waiting on. Based on an informal request to Chris at the August 1995 SHARE Technical Conference, IBM revised its CICS Problem Determination Guide (see above reference) to include a detailed itemization of the CICS waits. This invaluable information allows CPEXpert to provide a more comprehensive analysis of CICS delays.